

Institute for Visualization and Interactive Systems

University of Stuttgart
Universitätsstraße 38
D-70569 Stuttgart

Bachelorarbeit

Cross-device Multimodal Visualization for Wrist Rehabilitation

Pantelis Antoniadis

Course of Study: Data Science
Examiner: Dr. Tanja Blascheck
Supervisor: Fairouz Grioui, Xingyao Yu

Commenced: June 5, 2023
Completed: December 12, 2023

Abstract

Taking wrist rehabilitation sessions is a crucial part of the post-stroke rehabilitation process to foster recovery. The conventional exercises performed at rehabilitation centers with physiotherapists are called Range of Motion (ROM) exercises. They consist of a set of rotational movements around the wrist joint. During traditional rehabilitation sessions, physiotherapists use a Goniometer Ruler to measure joint limits and range of motion. Between the regular sessions with the physiotherapists, patients are asked to continue the exercises at home. However, they often lack motivation as well as the necessary guidance to perform the movements correctly, which can lead to postponing the recovery and may lead to serious problems. By using a smartwatch to address the issue of lack of support with at-home rehabilitation, a new problem arises: Guiding the patient during these exercises is problematic due to the occlusion of the smartwatch screen. Therefore, combining and extending the smartwatch screen with a smartphone or an augmented reality-based solution are potential options to overcome this problem. For this research two separate applications were built. The first application mirrors the visualization of the smartwatch to the smartphone, second is an AR application built to run on a smartphone, which is housed inside a Google Cardboard. To evaluate and compare the developed solutions, a study is conducted in which the three conditions will be analyzed.

Contents

1	Introduction	13
2	Background	15
2.1	Medical Background	15
2.2	Computer Graphics Background	16
2.3	Networking	17
3	Related Work	19
3.1	Smartwatch Visualization	19
3.2	Range of Motion Visualization	19
3.3	AR Visualization	20
4	Methodology	23
4.1	General Concept	23
4.2	Smartwatch App	23
4.3	Smartphone App	24
4.4	AR Application	26
5	Implementation	27
5.1	Configuration	27
5.2	Sending the data	28
5.3	Smartphone application	28
5.4	AR application	32
6	Study	37
6.1	Study Design	37
6.2	Participants	38
6.3	Apparatus and Materials	39
6.4	Procedure	41
7	Results	43
8	Discussion	49
8.1	SUS	49
8.2	NASA TLX	51
8.3	Preference Questions	52
8.4	Threats to Validity	54
9	Conclusion and Outlook	57
	Bibliography	59

List of Figures

4.1	The initial screen of the smartphone application	24
4.2	The screen on the watch while doing a rotation task	25
4.3	The screen on the watch presenting the summary of the exercise	25
5.1	Flow chart for displaying the data	29
5.2	Mirroring the visualization of the smartwatch on the smartphone	30
5.3	Screw and Solidify Modifier in Blender	32
6.1	Stereoscopic AR View	38
6.2	The Smartwatch and the Strap for the AR Condition	40
6.3	The QR-Strap and the Holder for the Smartphone	40
6.4	Participant during the AR condition using the Google Cardboard	42
7.2	Mean for NASA TLX on the first condition	46
7.1	Mean for the SUS on the first condition	46
7.3	Mean for SUS on the second condition	47
7.4	Mean for NASA TLX on the second condition	47
7.5	Preference Rating	47

List of Tables

6.1 Background Information of the Participants 39

7.1 Acronyms for Questions 45

List of Algorithms

5.1	Algorithm for Receiving the Initialization Parameters from the Watch	29
5.2	Displaying the Data on the Phone	31
5.3	Initialization for the Normal Scales and the Parameters for the AR Application . .	33
5.4	Updating the Visualization for the AR application	34
5.5	Receiving the data from the smartwatch in Unity	35

1 Introduction

Wrist rehabilitation takes an important role in the post stroke recovery. An indispensable facet of the rehabilitation is the practice of Range of Motion (RoM) 2.1.1 exercises, in which the patient has to perform rotational movements encompassing yaw pitch and roll around the wrist joint. Traditionally these exercises are monitored by physiotherapists using a Goniometer Ruler to measure joint limits and track the precision and range of the movements [NRM+18]. However, the rehabilitation does not stop at the rehabilitation centers. Patients are often instructed to perform and monitor the excises at home by themselves. Especially in the post stroke rehabilitation, hospitals tend to prefer early home-based inpatient treatment. It is more cost-effective for the hospitals and for the patients [RØK+16]. Because of that, the patients often have to do without an expert who can supervise their progress and guide them through the exercises that are recommended by the doctors. The absence of an expert to supervise the recovering process, can bring up a variety of challenges for many patients, as the lack of professional tools and knowledge makes the recovery process particularly difficult while still recovering. This can lead to a lack of motivation and can result in a longer or incomplete recovery.

With the evaluation of technology and the increased accessibility of it, the chance to close the gap between the clinical rehabilitation and at-home practice, presents itself. This research tries to find new ways for improving at-home rehabilitation with technology that is commonly used by many patients in their every day life.

To achieve that, this thesis presents the development of an extension for the smartwatch, which tracks the ROM of the wrist joint. The two extensions are based on a solution to mirror the visualization of a smartwatch to a smartphone and the use of AR, which can be experienced on a smartphone.

With the use of a traditional smartwatch at the disposal to track the rotation of the wrist, there are some immediate problems at sight:

- The occlusion problem when the wrist rotates, specifically on the x-axis
- The limited space to show important information regarding the ongoing process

With these prerequisites given the following research question arises, which this research aims to answer:

- **QR:** How can we overcome the occlusion problem of guidance visualization on a smartwatch during wrist rehabilitation using a smartphone

The solutions presented in this thesis include the development of a separate application for smartphones to mirror the visualization from the smartwatch onto the larger and more readable screen of a smartphone. Additionally, it includes an AR application that focuses on extending the smartwatch screen with technology that could become even more relevant in the future. Additionally, a study will be conducted to evaluate, analyze, and compare the different conditions.

The necessary background information for this thesis will be presented in Chapter 2. This includes both medical and computer graphics backgrounds. Chapter 3 will present previous work related to the research in this thesis, particularly concerning range of motion visualizations. Chapter 4 will focus on the tools and methods required for implementation. Following this, Chapter 5 delves into more detail about the technical aspects of the presented solutions. Chapter 6, concerning the conducted study, includes all details about the study design, materials used, and procedures. The information gathered from the study will be presented in Chapter 7 and discussed in detail in Chapter 8. To conclude, Chapter 9 summarizes the gathered information and provides an outlook on what is possible for future work, using this research as a basis.

2 Background

This section will give more insight regarding the approach of the two solutions developed in this work. It will also provide some background knowledge to why this study was conducted and explain some core ideas.

2.1 Medical Background

The following section will cover basic knowledge about the medical background which is associated with this research.

2.1.1 Range of Motion

The Range of Motion (ROM) is the distance an object travels while being attached to another. In our specific context, ROM refers to the angular distance and direction a joint can move between its fixed and extended positions. Traditionally, the ROM of the wrist is measured using a goniometer by a trained expert. Among other findings, Nizamis et al. used different goniometers to manually measure the angles of joint rotation [NRM+18]. Through therapeutic exercises (range of motion therapy), we aim to increase this distance.

2.1.2 Wrist Rehabilitation

After a wrist injury, it is recommended to perform rehabilitation exercises to improve the ROM. Tae Seob MD et al. explored the ROM of the wrist using an iPhone 4. According to their research, the normal values for wrist ROM are 73 degrees of flexion, 71 degrees of extension, 19 degrees of radial deviation, 33 degrees of ulnar deviation, 140 degrees of supination, and 60 degrees of pronation [TSM12]. In another study conducted by Than et al., the mean degrees of motion at the wrist were found to be 68.3 for flexion, 68.2 for extension, 19.6 for radial deviation, and 26.1 for ulnar deviation [KPL+14]. Based on these findings, this research uses approximate values for the maximum rotation of each axis.

The medical costs for hospitals have increased significantly in recent years, leading hospitals to implement cost-saving measures to reduce the costs of treating and maintaining patients. One such measure is the transition from inpatient to outpatient care, as described by Ferrier et al. [FLV09].

In our study, we specifically aim to solve the occlusion problem that occurs when trying to track the rotation of the wrist using a smartwatch. This is particularly relevant in the post-stroke wrist rehabilitation process for patients who are receiving home-based treatment.

2.2 Computer Graphics Background

2.2.1 Baking

The term baking is frequently used in computer graphics in regard to non-real-time rendering. While it is often associated with game development, it is not limited to this field alone. Baking is particularly useful for real-time applications that require complex rendering. Generally, baking can be distinguished into three different types[Ken23]:

- Texture baking
- Animation/Simulation baking
- Light baking

Although there are technical differences between these techniques, the general principle remains the same.

The significant advantage of texture baking is the transposition of complex 3D characteristics into a 2D texture. This process can include, but is not limited to, ambient occlusion, normals, or vertex colors. Additionally, lighting for a specific material can also be baked into a 2D texture, thus reducing real-time computational overhead. Consequently, this technique finds widespread use in game development. (Consider listing drawbacks)

Following the texture baking method, baking for animation and simulation also adheres to the principle of reducing calculation overhead during run-time. This involves replacing interpolation between keyframes, deformation of meshes, and different rigs within a hierarchy with a geometry sequence. This approach is significantly more efficient as the animation system changes, with the engine simply cycling through the sequence. Even more complex calculations, such as large-scale simulations including cloth, fire or smoke, can be baked, making them usable in a real-time rendering environment.

Light calculations in a scene can also be resource-intensive. Here, baking can be applied as well. For static lights, lightmaps are baked into a 2D texture, containing all the lighting information of a light object. In combination with dynamic lights, it is possible for objects to receive lighting information from these baked lightmaps. This approach is highly efficient in real-time calculation compared to the naïve method.

2.2.2 Vertex Animation Texture

The Vertex Animation Texture (VAT) technique incorporates the principle of animation baking. It bakes the animation of a 3D model into a texture map, which can then be utilized and morphed by various shaders. In this technique, the animation is completely handled by the shader. For VAT to work effectively, the local space vertex of each frame is stored in a matrix, as well as the normals. The normals are stored in a separate matrix because they differ from the vertices in each frame. This information is then stored inside a texture. A significant advantage of this approach is that the mesh of the 3D model remains static in the CPU. For example, in Unity, the engine is not aware that the vertices are being moved by the GPU.

2.2.3 Alembic

Alembic(abc) is a computer graphics file format developed by Sony Pictures Imageworks and Industrial Light and Magic [ale10]. It was first introduced at SIGGRAPH in 2011. The primary purpose of this file format is to facilitate the interchange of animated geometry among multiple collaborators. Abc supports commonly used geometric representations such as polygon meshes and subdivision surfaces. The format transforms animated scenes into a non-procedural set of baked geometric results, making it highly useful for baking complex animated scenes, for example, for the purposes of lighting and rendering.

2.3 Networking

The User Datagram Protocol (UDP) was first introduced by Postel in 1980 [Pos80]. It provides a method designed to send messages with minimal protocol overhead. UDP is a transaction-oriented protocol, and neither delivery nor protection against duplicate messages is guaranteed. This makes it especially useful for file transfers where speed is prioritized over guaranteed delivery. To use UDP, the Internet Protocol (IP) is required as an underlying protocol. In the context of this research, these constraints are not drawbacks, making UDP the preferred choice for data transmission.

The header of the protocol consists of:

- Source port
- Destination port
- Length
- Checksum

The source port is 2 bytes long and is used to identify the port number of the source. Similarly, the destination port, also 2 bytes long, identifies the port for the destined packet. The length field specifies the total length of the datagram, including both the header and the actual data, with the minimum length being 8 bytes. The checksum, 2 bytes long, is generated from a pseudo-header, the actual UDP header, and the data itself. The pseudo-header, placed on top of the UDP header, is necessary for transmission via IP.

After introducing the relevant background, this thesis will examine related work that inspired the approach to this research question.

3 Related Work

This section focuses on previous work that investigated the virtual extension of user interfaces and other methods that support the wrist rehabilitation process. We will review and analyze the methods being used in those researches.

3.1 Smartwatch Visualization

In their work, Wenig et al. introduced an extension for the smartwatch display called, WatchThru [WSO+17]. The extension is achieved through an attachable semitransparent display on the smartwatch. Through this display additional content can be displayed as a floating visualization over the main screen of the watch. Three interaction techniques are introduced: Pop-up Visuals, Second Perspective and Peek-through. The Pop-up Visuals extend the visuals out of the main screen. For instance incoming notifications can be viewed through the additional semitransparent screen. For the Second Perspective technique, the additional screen acts as an alternative view of the visualization on the main screen. A use-case for a second perspective could be, the 3D visualization of a 2D map. In the last of the presented techniques, Peek-Through, the additional display makes it possible to augment real life objects. This is in particular interesting because it uses AR-technology without the need to wear additional head-mounted displays.

3.2 Range of Motion Visualization

Kalime et al. developed a glove designed to translate exercises into interactive gaming actions, aiming to keep users more motivated [KAGE11]. Due to the difficulty in tracking the human's arm movement in the air, this research immobilizes the patient's arm on the x and y axes. An accelerometer, placed on the glove, measures the pitch (y-axis rotation) and roll (x-axis rotation). The data from the accelerometer is fed into an A/D converter, which outputs to the Sensory Data Module. This module is responsible for calculating the acceleration of each axis and the angles for pitch and roll tasks. A Communication Module is then used to relay the calculated angles to the game. The Game Controller Module translates these angles and velocities into activities and events within the game. To further increase patient motivation, multimedia elements like music or haptic feedback are integrated.

In their research, Ananthanarayan et al. developed a wearable knee pad to visualize and track knee movement, specifically during bending [ASC+13]. The knee pad's purpose is to assist patients at home with knee rehabilitation. It consists of an enclosure on the upper thigh housing the electronics, an enclosure for the calf, and a band sensor. The enclosures are made of curved Neoprene, lined

with polyester on one side and Spandex on the other. The band sensor, also made of Neoprene, connects to the calf and thigh enclosures. This solution supports full knee movement and does not hinder participants during exercises.

Following the principle of a serious game, Deponti et al. developed a smartphone application, Droidglove, which incorporates game-like features such as exercise goals and tracking of previous attempts [DMP09]. In a rehabilitation scenario, the patient holds their smartphone in hand, and the application tracks all the rotations of the prescribed exercise. A vibration notifies the user upon exercise completion. The application has five main functionalities: Motion recording, motion performing, video capturing, video playing, and data managing. Both motion recording and motion performing read data from the sensors, with motion recording validating the received data against stored data to ensure correct exercise performance. Video capturing enables doctors to record instructional videos, and video playing allows patients to view these videos. Finally, the data managing module handles stored exercises, retrieves usage statistics, and authenticates doctors, granting them access to a reserved area.

3.3 AR Visualization

Da Gama et al. [DCF+16] presented an AR rehabilitation system that utilizes Kinect skeleton tracking technology. This system tracks the skeleton model of the patient, allowing for the monitoring and analysis of their physical movements in real-time. A new movement recognition method was developed to recognize and classify biomechanical movements. Based on these recognized movements, the system can assess the patient's rehabilitation progress and provide a variety of exercise instructions designed to motivate them.

In their work, Debarba et al. introduced a visualization tool for human motion analysis using AR. The AR interface visualizes joint movements of the human body by rendering bones as a holographic overlay on the patient, enabling real-time tracking of their movements. This application was developed for the Microsoft HoloLens, which supports Optical See-through technology. Unlike other solutions, it synchronizes the data from the HoloLens's built-in motion tracking system with an external system, the VICON MX-T40s motion capturing system. This was crucial because, although the HoloLens has sensors for position and orientation tracking, it is not optimal for moving backgrounds. Thus, a transformation from the marker-based coordinate system of the VICON to the physical center of the HoloLens projection was necessary.

In another study, Aung et al. present the development of a rehabilitation system with biofeedback (RehaBio) for upper-limb rehabilitation using AR [AA14]. The RehaBio system consists of three major modules: the database module, the rehabilitation module, and the biofeedback module. The database module, developed on the MATLAB platform, provides background information about patients and their rehabilitation progress. The rehabilitation module offers exercises based on an AR approach. The biofeedback module tracks the patients' muscle performance and activities. The implementation of the exercise module and the AR approach is carried out on the Adobe Flash Professional platform, using the ActionScript API for capturing video scenes, displaying virtual objects, and searching for the marker that participants move with their hand. Only a webcam is required to recognize these markers. In a typical task, patients are required to catch virtual balloons as an exercise. The collision between the virtual balloon and the marker on the patient's hand is recognized by the exercise module.

This section concludes the review of related work for this research and sets the stage for the methodology chosen based on the previous studies presented.

4 Methodology

This section focuses on the methods and tools required for the development of the applications and the conditions of the study.

Our experimental testing environment was constructed using a Smartwatch application developed in the Java programming language. This application was designed to run on WearOS, specifically Android version 9. For our experiments, we employed the Fossil Carlyle HR 5135 smartwatch as the hardware platform. The implementation on the smartphone side was conducted on a Google Pixel 4a, running Android version 10.

To address the issue of occlusion, we implemented two distinct methodologies. The first approach is based on more conventional practices, involving the real-time synchronization of rotational data through mirroring visualizations from the smartwatch to a connected smartphone.

In contrast, the second approach incorporated an AR solution, aimed at exploring possibilities beyond traditional techniques. This AR-based solution was implemented on an Android 11 platform and tested on a Google Pixel 4a smartphone. The target SDK (Software Development Kit) version used for the application was set to version 31.

4.1 General Concept

The default application programming interface (API) provided by Google primarily supports data transfer between connected devices via Bluetooth. However, in our research, we encountered significant latency issues with this method, rendering it unsuitable for our objectives. Due to these limitations, we decided to employ the UDP for data transmission, utilizing a WiFi connection, a decision supported by empirical evidence of improved transfer rates and results.

By adopting the UDP protocol over WiFi, we successfully achieved real-time data transfer from the smartwatch to the smartphone with negligible delays. It is important to note that this method requires both the smartwatch and the smartphone to be connected to the same network. Therefore, this approach may be less practical in scenarios that demand greater mobility. Nevertheless, this method of data transfer was utilized in both of the solutions we developed.

4.2 Smartwatch App

At the center of the data transmission process is the smartwatch. Equipped with built-in accelerometer and gyroscope sensors, it can record the movement of the hand. To track all rotation axes effectively, the watch is worn on the back of the hand instead of the wrist.

The smartwatch app consists of three stages:

- The initialization Figure 4.1
- The rotation condition Figure 4.2
- The result summary Figure 4.3

In the initialization stage, the patient can select a set of axes they wish to exercise. Additionally, it is possible to adjust the number of rotations per axis. The initialization button, specifically built for this research, allows the smartwatch to synchronize parameters with the phone for the exercise. This feature was not present in the initial build of the application.

The visualization during the exercise consists of a radial chart and a combined bar chart. The radial chart displays in real time the angle currently achieved by the patient. The combined bar chart shows the results of each rotation with respect to the current axis, including the speed of the rotation.

At the end of the session, participants receive a summary of their performance. For each axis, an average of the maximum rotation and the rotation speed will be displayed.



Figure 4.1: The initial screen of the smartphone application

4.3 Smartphone App

To overcome the occlusion problem, we mirrored the visualization of the smartwatch on a smartphone screen. This allows the patient to access information about their exercise at any given moment. As previously stated, a separate smartphone application was developed. It receives angle data, as well as velocity and time for each rotation in real-time, allowing for full synchronization of the visualizations on both screens.

The development of the smartphone application was completed entirely in Android Studio and written in Java. One of the initial challenges was finding a suitable method for data transmission.



Figure 4.2: The screen on the watch while doing a rotation task



Figure 4.3: The screen on the watch presenting the summary of the exercise

Our first attempts to transmit data via Bluetooth were insufficient. The Bluetooth connection's transmission rate could not handle the volume of data from the smartwatch sensor. For this reason, the choice was a UDP solution, which allowed for a much higher transmission rate than the regular Bluetooth connection.

The smartphone app's visualization was adapted from the smartwatch, with a few modifications to fit the larger smartphone screen.

4.4 AR Application

Additionally, a separate AR application was developed, primarily to explore possibilities for extending the smartwatch screen, particularly in the context of wrist rehabilitation.

This application was built for Android and written in the C# programming language. It included coding for the behavior of prefabs and data transmission scripts.

The AR application was developed in Unity and built for Android OS smartphones. We used Google Cardboard to view the visualization with the smartphone.

Choosing Google Cardboard for the AR solution was based on two main design decisions. Firstly, it makes the solution more accessible to a broader audience due to the cost-effective nature of Google Cardboard, lowering the entry barrier. While dedicated AR hardware offers additional computing power useful for other use-cases, our visualization's relative simplicity meant this extra performance was unnecessary.

Secondly, the flexibility in development that comes with using a Google Cardboard was beneficial. It allowed for the use of the same phone for both the Smartphone and the AR condition, eliminating the learning curve of an additional development environment and saving time that was instead used to polish technical aspects of the implementation.

Given these considerations, the choice of Unity as the engine was evident. Unity is a sophisticated and widely-used engine in 3D animation and simulation, supported by a large community and comprehensive documentation, which is invaluable for bug-fixing and troubleshooting during implementation.

For the AR visualization, the projection was achieved with the Vuforia extension in Unity. We used a strap with two QR codes as the image target, placed on the participants' wrists. Data transmission was also handled via UDP.

To create the animation for the AR visualization, a 3D model and its respective animation were created in the 3D creation suite, Blender. The animation was then baked and exported as an abc file. As Android OS does not natively support abc file playback, making it impossible to build the application as an Android Package (apk) directly from Unity, we used a custom script to convert the abc file to a Vertex Animation Texture (VAT) [Gax20]. This approach enabled playback and rendering of the animation on the phone.

With the methodology outlined, the next chapter will provide a more detailed view of the implementation.

5 Implementation

Following the description of the General Concept, this chapter will focus on the implementation for the applications, which will help to evaluate the research question of this thesis. Additionally, it will give more insight and further details on how the results are achieved.

5.1 Configuration

Beginning with the configuration, there are various things to consider before the actual implementation can start.

5.1.1 Android Studio

In Android Studio, it is essential to install and update the Android SDK Command-line Tools as well as the Android SDK Platform-Tools, most importantly for the use of the Android Debug Bridge (ADB). ADB is a command-line tool that is crucial for building and running the application on the phone while enabling debugging.

One of the most important features for the use-case of this research is the ability to connect to and debug the device over Wi-Fi. This was necessary because the smartwatch used in this research did not support debugging over a USB cable. To facilitate this, the development settings on both the smartphone and smartwatch needed to be enabled. This procedure allows for the activation of Wi-Fi debugging in the watch's settings. It is also important to ensure that all devices are connected to the same network for successful communication.

Another potential solution to this challenge is the use of the device emulator that comes with Android Studio. However, since the application for this research heavily relies on actual movement with the smartwatch, debugging over Wi-Fi on the actual device is preferred. Additionally, the resources required to run an emulator can lead to frustration during development, further supporting the choice against this approach.

5.1.2 Unity

After configuring Android Studio, several additional settings need to be adjusted within the Unity Engine as well.

To build the application for Android OS, the build platform must be changed in the build settings. The minimum Android SDK target should be set to Android 7 (API level 24).

For image-target recognition, the Vuforia Engine SDK is required. This SDK must be manually imported into Unity. After the import, a Vuforia AR camera can be added to the scene within the engine. To do this, an App license key is required, which can be obtained from the official Vuforia website. This key must then be entered in the Vuforia configuration inside Unity. To create an image target, a database with the chosen image must be created on Vuforia and then imported into Unity.

There is also a potential solution for implementing the stereoscopic view required for the AR application, involving the Google Cardboard SDK, which can be imported via the GitHub repository. However, after some testing, compatibility issues with the Vuforia engine arose, leading to a custom solution that involved using a separate camera for each eye.

To process the animation, two more packages are needed. The abc package must be imported into Unity to enable the processing of abc files. Additionally, the Unity-AlembicToVAT project from GitHub is required to convert the abc file into a Vertex Animation Shader.

With these configuration steps, the development environment is prepared for the actual implementation, which will be discussed in the next section.

5.2 Sending the data

The first functionality that needs to be implemented on the smartwatch is the data transfer between different devices. In Java, this can be achieved using the Datagram class. The initial step is to initialize the network. On the receiving end, this includes setting up a receiving port and defining a buffer length for the packet. For sending data, we need to specify a destination IP address, a destination port, and also a buffer length for the UDP packet. The parameters sent from the smartwatch are combined into a string, separated by commas. These parameters include:

- 'Current Axis Status,' which indicates which axis is currently in use.
- 'Test Status,' indicating whether a test is currently ongoing.
- 'Repetition Parameter,' defining the selected number of rotations for a test.
- 'Active,' indicating whether a rotation is currently ongoing.

5.3 Smartphone application

The initial layout of the application features two buttons. One button is used to launch the mirroring visualization, and the other is for its initialization. After creating an initial layout in Android Studio, two methods for receiving data are implemented. These methods handle the reception of both the data and the initialization parameters, respectively. In this context, the initialization parameter refers to the number of rotations selected on the smartwatch. This setup allows for the passing of these parameters to create combined charts in the subsequent layout.

To visualize the received data, a new layout is created, which can be accessed through the corresponding button on the initial layout. For visualization purposes, the smartphone application utilizes the same library as the smartwatch.

Algorithm 5.1 Algorithm for Receiving the Initialization Parameters from the Watch

```

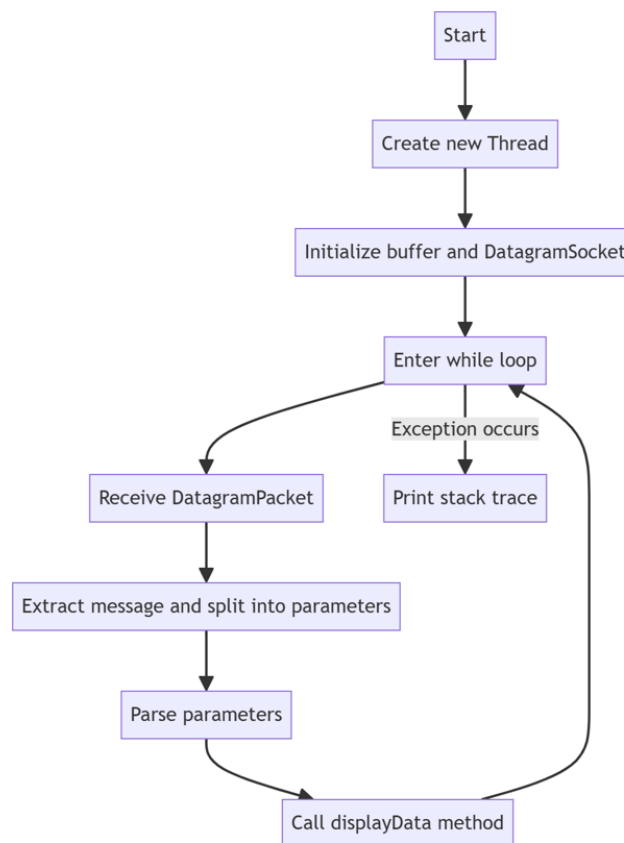
public void initialReceive (int portNum) {
    initial = new Thread() {
        @Override
        public void run() {
            try {
                final int taille = 8192;
                final byte[] buffer = new byte[taille];
                DatagramSocket socketReceiveInit = null;
                socketReceiveInit = new DatagramSocket(portNum);
                Log.d("initDatagram", "created");

                while(true) {

                    Log.d("initwhile", "enteredWhile");
                    DatagramPacket datainit = new DatagramPacket(buffer, buffer.length);
                    socketReceiveInit.receive(datainit);
                    String message = new String(datainit.getData(), datainit.getOffset(),
                    datainit.getLength());
                    repetitionPara = Integer.parseInt(message);
                    Log.d("initialpara", "received" + repetitionPara);

                }
            }
        }
    };
}

```

**Figure 5.1:** Flow chart for displaying the data

5 Implementation

In Figure 5.1, the diagram showcases the control flow for receiving and displaying data from the watch. Once in the visualization layout, a new thread is created to house the entire logic. A new Datagram Socket is initialized to receive the actual data from the smartwatch. The program then enters a while loop that receives a packet, extracts the parameters from the parameter string, and parses them into variables. Finally, the method to display the data is called.

To display the received data from the smartwatch on the smartphone, the visualization on the smartphone updates with each new rotation received. This ensures that the radial chart on the phone is in sync with its counterpart on the watch.

Here is a breakdown of the control flow of the displayData method: First, the incoming angle data from the watch is parsed into a float. The next step is to check the current test status, which indicates whether the user has completed their tasks. If a test is ongoing, then the current axis is identified. Depending on the direction of rotation, the angle chart is updated and checked to see if the maximum angle of its corresponding axis is reached. Once the current angle returns to zero, the angular velocity and test time are calculated. The results of the completed rotation, along with the test time and angular velocity, are then displayed on the record chart on the left side of the smartphone screen.

An important aspect to note is that receiving and displaying the data must be done on a separate thread. This ensures that these actions do not block the main thread of the application, as an ongoing process on the main thread could cause the application to crash.

Similar to the application used in the first condition, the AR application also relies on the same protocol for data transfer. Consequently, the implementation in Unity is similar, with one of the few differences being that the programming language used was C#. It is crucial to ensure that the applied script remains active at all times. This is particularly important when the camera loses focus on the image target; the DatagramSocket should still be capable of receiving messages. If the focus is regained, the visualization will then be at the desired stage. In this case, it was sufficient to apply the script to the AR camera

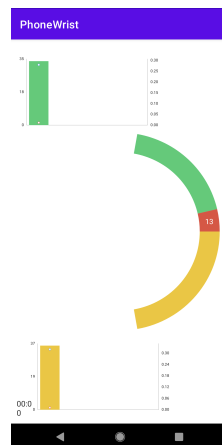


Figure 5.2: Mirroring the visualization of the smartwatch on the smartphone

Algorithm 5.2 Displaying the Data on the Phone

```

    if (rotateDirection) {

        if (diff >= 0)
            rotationChart.setData(90 - leftDefaultMaxAngle, leftDefaultMaxAngle, Math.min(diff,
            rightDefaultMaxAngle), Math.max(rightDefaultMaxAngle - diff, 0), 90 - rightDefaultMaxAngle);
            checkAchievedMaxAngle(diff, rightDefaultMaxAngle);

    } else if (!rotateDirection) {
        if (diff <= 0)
            rotationChart.setData(90 - leftDefaultMaxAngle, Math.max(leftDefaultMaxAngle - Math.
            abs(diff), 0), Math.min(Math.abs(diff), leftDefaultMaxAngle), rightDefaultMaxAngle, 90 -
            rightDefaultMaxAngle);
            checkAchievedMaxAngle(diff, leftDefaultMaxAngle);
    }

    if (!active) {
        stopwatch.startStopwatch();
    }
    barupdate = true;
} else {
    if (active) {
        stopwatch.stopStopwatch();
    }
    int index = rotateDirection ? 1 : 0;

    testTime[index][1] = stopwatch.getTime() - testTime[index][0];

    caculateAngularVelocity(index);

    directionIconImageView.setImageResource(0);

    if (barupdate) {

        if (rotateDirection) {

            RotationData_Phone rotationData = new RotationData_(axisStatusText, repetitionDone
            [index], testMaxAngle, testTime, testAngularVelocity);
            rotationDataList.add(rotationData);

            rightRecordChart.addData(testMaxAngle[index], (float) testAngularVelocity[index
            ][0], (float) testAngularVelocity[index][1]);

            rotateDirection = !rotateDirection;
            testMaxAngle[rotateDirection ? 1 : 0] = 0;
        } else {
            leftRecordChart.addData(testMaxAngle[index], (float) testAngularVelocity[index
            ][0], (float) testAngularVelocity[index][1]);

            rotateDirection = !rotateDirection;

            testMaxAngle[rotateDirection ? 1 : 0] = 0;
        }
        repetitionDone[index]++;
        test.interrupt();
        barupdate = false;
    }
}

```

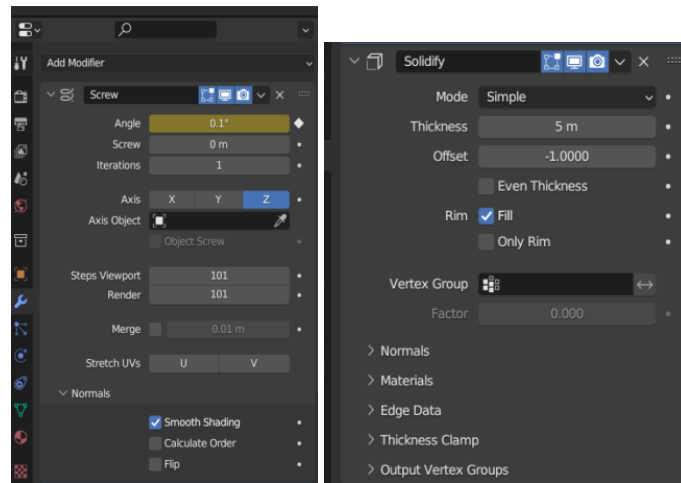


Figure 5.3: Screw and Solidify Modifier in Blender

5.4 AR application

Creating the Animation

To create the animation for the AR condition, the first step is to generate a 3D model using procedural modeling techniques in Blender.

A set of modifiers is applied to primitive geometry. In this case, two modifiers from Blender, namely the Screw modifier and the Solidify modifier, are applied to a pair of vertices. The Screw modifier creates circular geometry around the origin point, while the Solidify modifier allows for the extrusion of faces in the direction of their normals, resulting in the desired ring model. To animate the ring's circular motion, additional steps are required. By setting keyframes at the start and end points of the Screw modifier angle, a linear interpolation between these points creates the desired rotational animation.

Due to Blender's use of modifiers, an additional step is required for Unity compatibility. Unity does not support the procedural algorithms of Blender's modifiers. To address this, the animation must be baked and exported. This is achieved by exporting the model as an abc file. By setting the number of keyframes, the animation is baked into a geometry sequence that can be imported into Unity. In this specific case, the number of keyframes, and thus the animation steps, is set to 100. This count allows for a more readable representation in percentages, facilitating easier understanding and manipulation.

Moving forward, the 3D model exported from Blender, including the corresponding animation, is imported into Unity. Once in Unity, it is essential to transform the abc file into a Vertex Animation Texture (VAT). This process is made possible with the help of the 'AlembicToVAT' package [Gax20].

Algorithm 5.3 Initialization for the Normal Scales and the Parameters for the AR Application

```
private Material _material_shader;  
  
private Vector3 newscale = new Vector3(1, 1, -1);  
  
private Vector3 normalscale = new Vector3(1, 1, 1);  
  
public float angle;  
public int axisStatus;  
public float maxAnglePositive;  
public float maxAngleNegative;
```

Controlling the Animation

With the 3D model and animation in Unity, the next step involves creating a script to control the animation based on the current rotation of the wrist. The created script for the animation control, must be attached to the imported 3D model. The rotation information is passed through from the AR-Camera, which houses the script for receiving the data. With this setup, the model's time position can be adjusted. To achieve this, calculations are performed based on the current angle, seen in algorithm 5.4.

The time position in the animation ranges from 0 to 1, while the value from the smartwatch represents an angle from 0 to the maximum angle of the current axis. The script accounts for this by checking whether the received value is positive or negative, indicating the rotation's direction. It then normalizes the value against the maximum positive angle for positive rotations or the maximum negative angle for negative rotations.

It's also important to adjust the scale of the 3D object normals for the opposite direction. Therefore, the sign of the z-coordinate of the 3D object is changed, thus mirroring the 3D object on its z-axis, ensuring that the normals face the opposite direction during negative rotations. Algorithm 5.3.

To enable viewing the application's content with the Google Cardboard, a stereoscopic view of the camera must be implemented in Unity. The camera's signal is captured and projected onto a plane with a resolution of 1920x1080, matching the resolution of the smartphone screen used for this condition. The plane is divided into two halves, allowing the same camera signal to be projected onto both parts. This setup creates the required stereoscopic view for the condition.

This concludes the implementation of the developed tools, and the study can now be conducted. The study will evaluate the presented solutions, providing more insight into their usability and effectiveness.

5 Implementation

Algorithm 5.4 Updating the Visualization for the AR application

```
void Update()
{
    GameObject angles = GameObject.Find("ARCamera");
    UDPListener listener = angles.GetComponent<UDPListener>();
    maxAnglePositive = listener.maxAnglePositive;
    maxAngleNegative = listener.maxAngleNegative;
    angle = listener.angle;

    if (angle < 0)
    {
        transform.localScale = newScale;
        angle = angle * - 1;
        angle = angle / maxAnglePositive;
        if (angle < (1))
        {
            _material_shader.SetFloat("_Timeposition", (math.round(angle *1000)) / 1000);
        }
        else if (angle > 1)
        {
            angle = 1;
            _material_shader.SetFloat("Timeposition", angle);
        }
    }
    else
    {
        transform.localScale = normalScale;
        angle = angle / maxAngleNegative;
        if (angle < (1))
        {
            _material_shader.SetFloat("_Timeposition", (math.round(angle *1000)) / 1000);
        }

        else if (angle > 1)
        {
            angle = 1;
            _material_shader.SetFloat("Timepostion", angle);
        }
    }
}
```

Algorithm 5.5 Receiving the data from the smartwatch in Unity

```
private void ReceiveMessage()
{
    // Check if data is available
    if (udpClient.Available > 0)
    {
        Debug.Log($"entered receive");
        try
        {
            // Receive data
            IPEndPoint remoteEndPoint = new IPEndPoint(IPAddress.Any, 0);
            byte[] data = udpClient.Receive(ref remoteEndPoint);
            message = Encoding.UTF8.GetString(data);
            string[] paraList = message.Split(",");
            angle = float.Parse(paraList[0]);
            axisStatus = int.Parse(paraList[2]);

            switch (axisStatus)
            {
                case 0:
                    maxAnglePositive = 80;
                    maxAngleNegative = 80;
                    break;
                case 1:
                    maxAnglePositive = 70;
                    maxAngleNegative = 90;
                    break;
                case 2:
                    maxAnglePositive = 50;
                    maxAngleNegative = 20;
                    break;
            }
        }
        catch (SocketException e)
        {
            Debug.LogError($"Socket Exception: {e}");
        }
    }
}
```

6 Study

The following chapter will go over the thought process that went into the creation of the questionnaire and the methodology behind it. The conducted study will help to evaluate and give a potential answer to the research question of this work.

6.1 Study Design

Due to the involvement of human subjects in this study, four basic ethical principles were applied: informed consent, scientific value, beneficence, and confidentiality [VS08]. Adhering to these principles ensured that participants were not harmed and were fully informed about the collection and storage of their data.

For evaluation purposes, we conducted a qualitative study using the NASA Task Load Index (NASA TLX) questionnaire, as introduced by Hart and Staveland (1988) [HS88], and the System Usability Scale (SUS) introduced by Brooke (1995) [Bro95].

The study involved a total of 10 participants, each of whom completed a comprehensive background form. This form helped ensure a diverse participant pool, representing various age groups, occupations, and environmental contexts.

Throughout the study, participants were instructed to perform identical tasks across the first two conditions. Each condition required the completion of five iterations for each axis, amounting to a total of 30 rotations per condition. For the AR condition, participants were asked to test the visualization and provide their feedback on the solution. This was necessary because the AR implementation did not offer the same level of information as the other conditions, making a direct comparison difficult using the same questionnaires.

Apart from the questions of the SUS and the NASA TLX, the open questions posed to the participants were as follows:

- Do you think that the AR visualization is helpful?
- Did you feel uncomfortable during the task?
- Does this approach motivate you more to complete the task?
- Do you think that the AR visualization is enjoyable?
- What did you not like about this condition?
- What improvements do you suggest for this visualization approach?

The study consisted of three conditions:

- First condition: Smartwatch only
- Second condition: Smartwatch and the Smartphone
- Third condition: Smartwatch and AR

6.1.1 Concluding Questions

To conclude the study, participants were asked a few questions regarding their personal preferences among the conditions. They made their choices based on their experience with each solution. These questions include:

- Which condition did you prefer the most?
- Which condition was most enjoyable?
- Which condition was most helpful?
- Which condition was most practical?
- Which condition do you think is more suited for the use at home

The last section of the questionnaire is reserved for participants to share their final thoughts and suggestions for improvements.

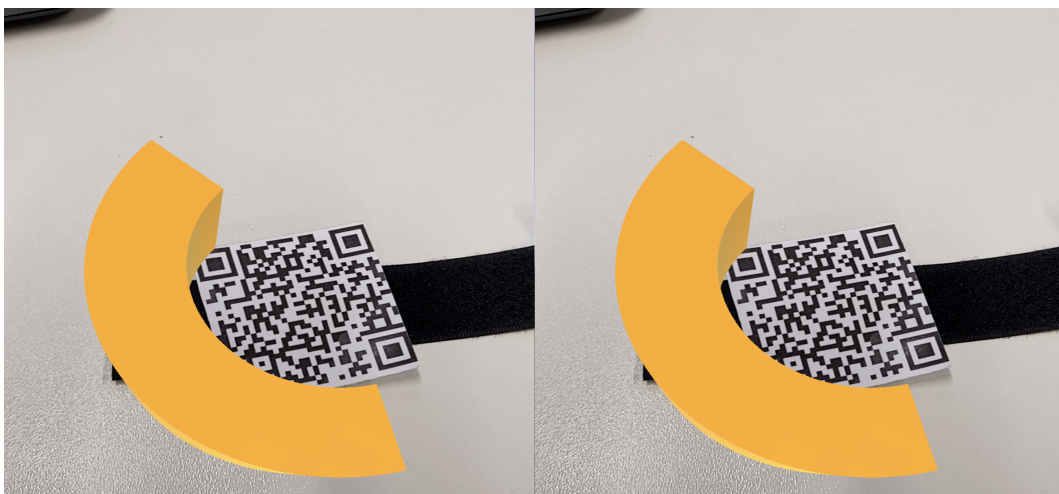


Figure 6.1: Stereoscopic AR View

6.2 Participants

In the conducted study, a total of 10 participants completed the process. These participants were recruited via email invitations and personal contacts, which resulted in a large portion being students. Only one participant is currently working in the industry. The ages ranged from 23 to 30, with the median age being 26 years.

Four participants had no prior experience with AR products, and three had undergone physiotherapy procedures in the past. These physiotherapy sessions involved the knee and the back. However, there was no experience with wrist rehabilitation among the participants in this study.

Table 6.1: Background Information of the Participants

PID	Age	Occupation	Experience with AR	Gone through physiotherapy?	Physiotherapy Details
1	25	Industry	Yes	Yes	Knee rehabilitation
2	26	Student	Yes	Yes	Back rehabilitation
3	25	Student	No	No	-
4	23	Student	No	No	-
5	27	Student	Yes	No	-
6	26	Student	Yes	Yes	Knee rehabilitation
7	26	Student	Yes	No	-
8	25	Student	Yes	No	-
9	27	Student	No	No	-
10	30	Student	No	No	-

6.3 Apparatus and Materials

The tool chosen for constructing and publishing the questionnaire was SoSci Survey. It offered considerable flexibility with features like local data storage and a pretest functionality. The pretest feature made it possible to run a pilot study and gather useful, meaningful feedback from participants, thereby enhancing the process of creating a coherent questionnaire. In contrast to other tools such as Google Forms, SoSci Survey allows the use of larger scales when constructing questions. This was particularly beneficial for the NASA TLX scale, which requires a scale of up to 20 elements, a capability not possible with Google Forms.

For the second condition involving the smartphone, a smartphone holder was used. It was configured so that the participant could easily view the phone while rotating the watch. Additionally, a strap with two QR codes was used as the image target and placed around the participants' wrists.



Figure 6.2: The Smartwatch and the Strap for the AR Condition

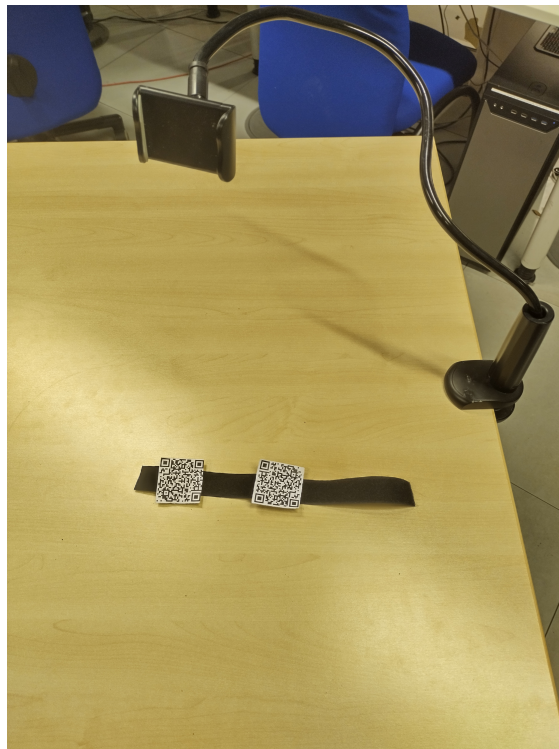


Figure 6.3: The QR-Strap and the Holder for the Smartphone

6.4 Procedure

Each participant was guided through the study individually. Apart from the participant, only one researcher was present during the session, which took place in the study room. Each session approximately lasted 45 minutes. After being briefed about the study procedure and its purpose, participants signed a consent form. This form included information about the procedure, potential risks, and the use of personal data.

After completing the background form, the study commenced. Participants performed 10 rotations on each of the three axes for each condition. Between conditions, the researcher had sufficient time to set up the next part of the study. Additionally, participants were given the opportunity to take a short break after each condition. No further explanation about the visualization details was provided beyond what was listed in the consent form. All tasks were performed using the left wrist.

6.4.1 Smartwatch App

The first condition for the participants involved using only the smartwatch. This included experiencing the occlusion problem, which was intended to familiarize them with it. Apart from the smartwatch, there was nothing else on the table or available for the participants to use. Before beginning the actual condition, there was a test round to help participants get accustomed to the procedure. After completing all the rotations on the specified axis, the participants filled out the questionnaire, which included the SUS and the NASA TLX scale.

6.4.2 Smartwatch and Smartphone

For the second condition, the participants used the smartwatch in combination with a smartphone. The smartphone holder was placed in front of the participants, and the smartphone was positioned inside the holder at an angle comfortable for the participant. After setting up, the participants were given instructions on what would be displayed on the smartphone. Although the rotation data was shown on the smartphone screen, the participants were required to tap on the watch to enable the next rotation. This setup allowed them to view the data of their task at any given moment while rotating their wrist. Similar to the previous condition, after completing this part of the study, participants filled out the same questionnaire.

6.4.3 Smartwatch and AR

In the final condition, participants wore the Cardboard mount with the phone attached to it. Before starting, the strap with QR codes was placed around the participant's wrist. After ensuring it was visible and properly positioned, the participant tried on the smartphone with the Google Cardboard. While receiving instructions from the researcher, the participant had the opportunity to test whether the visualization caused any discomfort. If any discomfort was experienced, there was an alternative option available that did not involve the stereoscopic view. In this scenario, the participant would hold the smartphone in their hand to view the visualization. In this study, however, all participants were able to use the solution with the Google Cardboard. As mentioned, the questionnaire for this condition differed from the first two.



Figure 6.4: Participant during the AR condition using the Google Cardboard

This condition marks an exception regarding the questionnaire that the participants filled out. The rationale behind this decision lies in the fact that the AR implementation's visualization does not provide the same level of detail as the previous conditions. Specifically, the combined charts present in earlier conditions are absent here, highlighting areas for improvement and further development in the future. This difference makes direct comparison of the results challenging. Consequently, the focus of the research shifted for this condition. Participants were asked open-ended questions based on their experience with the AR condition.

7 Results

This chapter will review all the data resulting from the questionnaires used in the study. Additionally, it will present the responses to the open-ended questions and some additional comments from the participants regarding the study.

The results of the first two conditions are represented using the median of all the answers given by the participants in their respective questionnaires. This means that for each of the two conditions, two graphs are presented. The median of all the SUS responses for the first condition can be seen in Figure 7.1. Similarly, the median for each question of the NASA TLX is shown in Figure 7.2.

Figure 7.3 and figure 7.4 display the means for the questions of the SUS and NASA TLX for the second condition.

For better readability in the graphs, the acronyms of the questions are listed in Figure 7.1. The results for the concluding questions about the participants' preferences are presented in a separate graph, shown in Figure 7.5. Some important and interesting responses to the open-ended questions regarding the last AR condition are also present in this chapter.

Regarding the open-ended questions for the AR condition in the questionnaire, below are some responses provided by the participants. Along with each response, the corresponding Participant ID is listed:

Some notable responses to the question, 'Do you think that the AR visualization is helpful?' were:

- ID 6 : "It was not more helpful but more pleasing to the eye"
- ID 7: "The AR visualization is not necessary more helpful for the physiotherapy exercise, but it is more improved than the radial green and yellow charts."

Only two out of the ten participants found the visualization to be helpful.

In contrast, three participants stated that the AR visualization did not cause them any trouble. However, some comments were made about the uncomfortable feeling experienced during the condition. These comments followed the question: 'Did you feel uncomfortable during the task?' The responses were:

- ID 7: "Yes. As an owner of a VR headset with pass-through the AR view is too much different from the reality. The view seems too much zoomed in, and it looks like you would look through binoculars. Because of this my sense of reality and what I was seat did not match, and it gave me some vertigo."
- ID 8: "Yes a little. The setup could have been a little better. With a better setup the uncomfortable feeling would not be as noticeable."

7 Results

In response to whether the AR approach increased their motivation to complete the task, two participants stated that it did not make any difference regarding their motivation. Some of their reasoning included:

- ID 7: “Yes, I feel more immersed and because of the visualization it motivates me more to complete the task. Because the donut gets filled completely in comparison to the radial/bar charts I get a better sense that the task is completed.”
- ID 8: “No, I think it did not make any difference compared to only the watch and in combination with the smartphone application.”

Regarding the enjoyability of the AR visualization, 2 participants felt that it was not enjoyable. In contrast, 8 participants found that both the visualization and animation were enjoyable and pleasing to look at. Some of their responses included:

- ID 7: “Yes, I enjoyed the AR visualization, but the implementation could be better. The screen was a bit too dark in comparison to the room light and the view was too much zoomed in. A fish eye lens effect probably would have looked better.”
- ID 8: “No, the setup could have been better. The AR visualization would’ve been much more enjoyable this way.”

Asking them specifically what they did not like about the AR-visualization resulted into following answers:

- ID 6: “The dizziness after taking off the head-mounted display”
- ID 7: “I didn’t like the vertigo because the AR seemed to be distorted. Maybe better hardware would have solved this issue.”
- ID 4: “the 3D visualization occluded itself, the camera was too far zoomed in, making me feel a bit dizzy”
- ID 9: “The visualization was not so sharp”

The final question was about possible improvements. Some suggestions were:

- ID 3: “Arrows on the AR visualization that show which way you have to rotate your hand”
- ID 7: “The visualization could run smoother. Maybe the donut could also be rotated a bit to the user so that you see a tilted O shape instead”
- ID 4: “A 2D dimensional visualization would be more appropriate. The visualization should be different for different kinds of hand movement, i.e., show which hand movement is currently practiced.”
- ID 8 “More feedback when rotating the wrist and maybe some gamification to motivate more”

After completing the questionnaire, participants had the opportunity to provide additional comments regarding the study, visualizations, and suggestions for further improvements. Some of their comments included:

- ID 3: “Visualization on the watch is not possible to read when the wrist is rotated, Screening method is unintuitive as It’s hard to look away from the rotating wrist and focus, AR is the best approach to the problem but needs some visual improvements.”

- ID 4: “I did not understand the bar charts and the lines on them on the smartwatch and the phone. The graph visualization was not marked, and it did not convey me any information, so I did not understand what data I’m looking at. The color of the arrow on the smartwatch has not enough contrast and the symbol itself is a bit small, and you can’t see it very well. The radial charts on the smartwatch could be labeled L and R, though their placement would infer this, probably. I feel the phone condition could have more information than the smartwatch visualization. A better AR experience would enhance the study.”
- ID 5: “stats were not very self-explanatory, color-coding could be consisted in AR (left green, right yellow), maybe short introduction video/tutorial.”
- ID 2: “The figure in the end is not understandable and the AR functionality is not really necessary.”
- ID 4: “The diagrams are not self-explaining. Understanding them needs help. – Always having to press on the smartwatch is annoying. It should recognize on its own when a movement is finished and directly start the next one.”

Table 7.1: Acronyms for Questions

ID	Question
QS1	I think that I would like to use this system frequently
QS2	I think the system is unnecessarily complex
QS3	I think the system was easy to use
QS4	I think that I would need the support of a technical person to be able to use this system
QS5	I think the various functions in this system is well integrated
QS6	I think there is too much inconsistency in this system
QS7	I imagine that most people would learn to use this system very quickly
QS8	I think the system very cumbersome to use
QS9	I feel very confident using the system
QS10	I think i need to learn a lot of things before I can get going with this system
QT1	How mentally demanding was the task?
QT2	How physically demanding was the task?
QT3	How successful were you in accomplishing what you were asked to do?
QT4	How hard did you have to work to accomplish your level of performance?
QT5	How insecure, discouraged, irritated, stressed, and annoyed were you?
QP1	Which condition did you prefer the most?
QP2	Which condition was most enjoyable?
QP3	Which condition was most helpful?
QP4	Which condition was most practical?
QP5	Which condition do you think is more suited for the use at home?

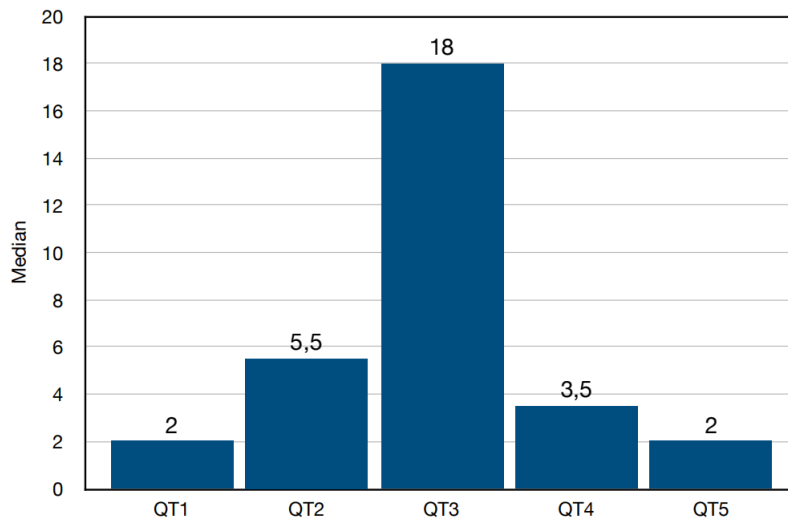


Figure 7.2: Mean for NASA TLX on the first condition

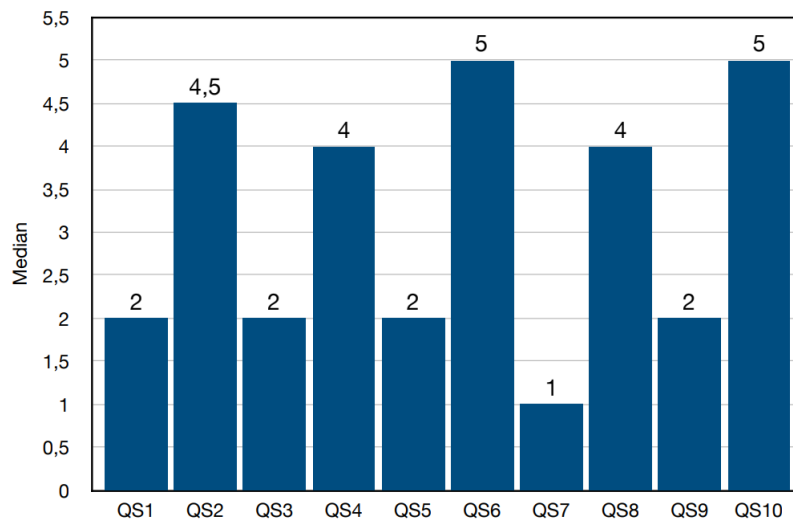


Figure 7.1: Mean for the SUS on the first condition

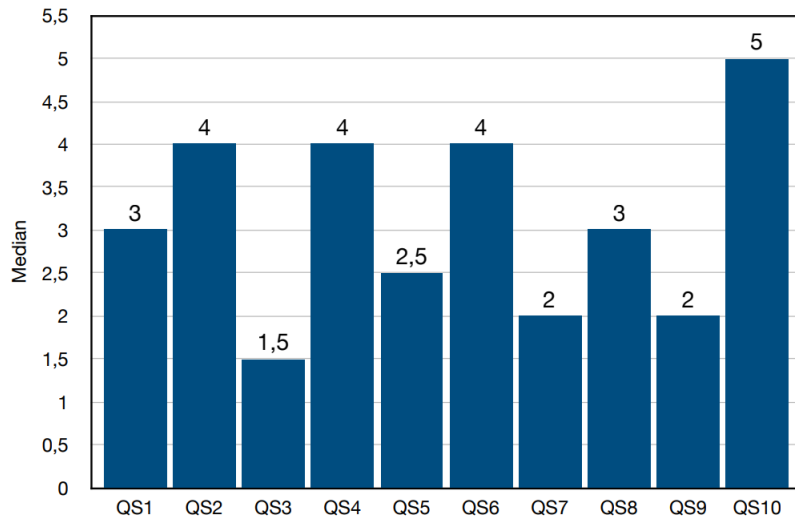


Figure 7.3: Mean for SUS on the second condition

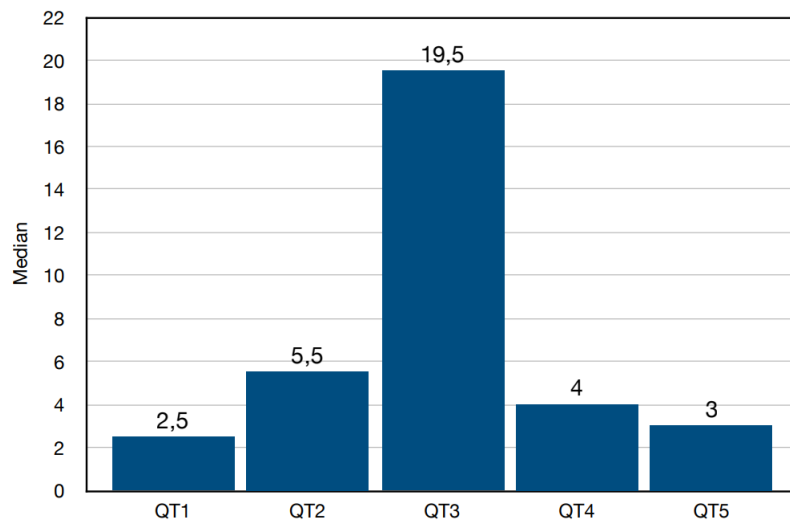


Figure 7.4: Mean for NASA TLX on the second condition

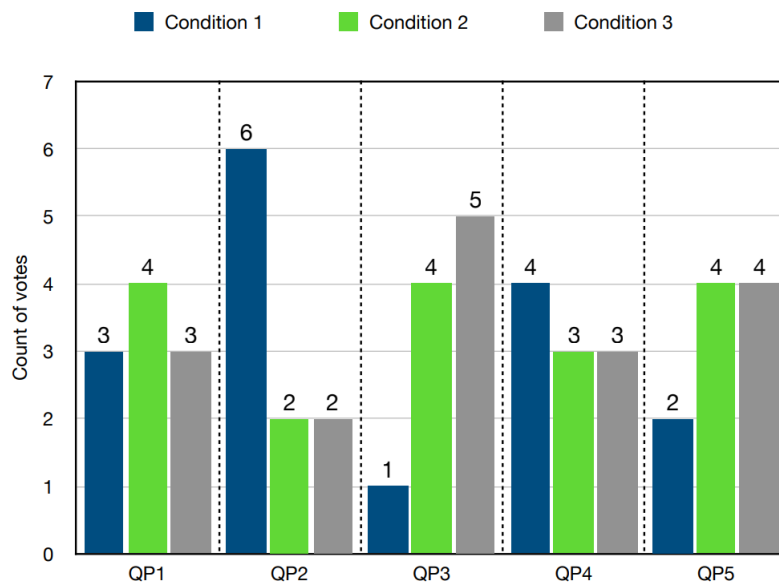


Figure 7.5: Preference Rating

8 Discussion

This chapter discusses the presented results of the conducted study and addresses the research question of this thesis. It is important to keep in mind that the SUS ranges from one to five. With one being rated as strongly agree and 5 being rated as strongly disagree. Accordingly, the answers of the NASA TLX, are ranging from twenty, which is representative for very high and zero, which represents a very low rating.

8.1 SUS

In this section the results of the SUS will be discussed.

QS1: I think that I would like to use this system frequently

Going from the first condition to the second, the median increased from two points to three. The added complexity could hinder the overall motivation to set it up and complete the exercises. This could indicate that the additional real-time information about the rotation of the wrist is adding too much complexity.

QS2: I think the system is unnecessarily complex

With an increase of 0.5 on the median for the second question the complexity of the second condition increases a small amount compared to the first one. With an added component in form of the smartphone screen, thus increasing the additional information given to the user, it does not seem to add much more to the perceived complexity by the user. Keeping that in mind, it is still an increase in complexity, which can also be derived from some comments of the users.

QS3: I think the system was easy to use

A difference of just 0.5 on the median in favor of the second condition is marginal. It seems to contradict the results of the question in 8.1. It is important to keep in mind that the added experience of the participants with the tasks could play a role in this result

QS4: I think that I would need the support of a technical person to be able to use this system

There was no difference in the mean regarding the need of a technical person to use the system. It is an indication that the added complexity is not too much to handle for the use at home. That being said, most of the participants had a technological background which can influence this result.

QS5: I think the various functions in this system is well integrated

The mean of the answers is increased by 0.5 which means that there is some improvement on the integration of the features to be done. One of the reasons for this result could be the need to tap on the smartwatch screen for each rotation and looking back on to the smartphone screen again. This was also mentioned by some participants in 7.

QS6: I think there is too much inconsistency in this system

Comparing the two conditions, the median decreases by one. The decrease translates to a bigger received inconsistency by the participants. This observation follows the same trend as in 8.1 and 8.1. Despite the decrease, the median is still not in a critical stage and the decline is not as severe as it might seem.

QS7: I imagine that most people would learn to use this system very quickly

The result of this question corresponds nicely with the assumption made in 8.1. Despite some increase in the complexity of the process, it is not enough to make the system hard to learn for new users. The decrease in the median indicates a better result for the second condition. With a difference of one, it could also be the Participants increased knowledge and practice on the task distorting the results.

QS8: I think the system very cumbersome to use

The median of both conditions is indicating a relatively convenient solution. In this case the added complexity seems to influence the perception of the Participants. With a decrease of one, the approval of the statement is weaker in the second condition. It is not a big difference and the result does not suggest a cumbersome system overall, but it is a trend that can be noticed across the overall results of the SUS.

QS9: I feel very confident using the system

This part of the questionnaire had no changes in the median between the two conditions. Both of them are not ideal with a median of two. There are some factors that could have played into this result, which are also mentioned by the participants in the open question section of the questionnaire. The occlusion problem of the watch, making it difficult to read the visualization, could lead to some frustration. Also, the fact that the application required a tab on the smartwatch from the user leads to uncertainty if the rotation is being tracked or not. Finally, the visualizations often were not understandable without any help 7

QS10: I think i need to learn a lot of things before I can get going with this system

Like in 8.1 the median of the second condition did not change compared to the first condition. At a median of five it is on the maximum of the scale. The participants were disagreeing with the statement, thus indicating that the system is not complex enough to require a big learning curve. This result also undermines the assumption that the added complexity was not enough to cause a disturbance in the perception of usability.

In conclusion, the conditions seem to perform similarly regarding the SUS. There are deviations between the two and the comments of the participants seem to underlay those results.

8.2 NASA TLX

In this section the results of the NASA TLX survey will be discussed.

QT1: How mentally demanding was the task?

The difference between the two conditions on the first question of the survey is marginal. Where the first condition had a median of 2 the second condition achieves a median of 2.5. The slight increase in the second condition could be correlated with the added smartphone component on the second condition. According to those scores, both conditions were not mentally demanding for the participants.

QT2: How physically demanding was the task?

No surprises here as all the participants did not have any form of wrist injury, which could cause any difficulties during the tasks. The median on both conditions was 5.5.

QT3: How successful were you in accomplishing what you were asked to do?

The median on both conditions is as expected high. A median of 18 on the first condition and 19 on the second condition. The slight increase might be caused by the additional visual feedback that is presented during the second condition. The visual feedback during the rotation on the smartphone could lead to a slightly bigger feeling of accomplishment. To overcome the occlusion problem this aspect is significant.

QT4: How hard did you have to work to accomplish your level of performance?

In similar ways as in 8.2, the answers of the participants follow the same trend. With a median of 3.5 for the first condition and 4 for the second condition, the scores are very low. Additionally, the insignificant increase of the mean in the second condition also shows that the implementation and overall design of the second condition, did not hinder the participants in concluding the rotation tasks.

QT5: How insecure, discouraged, irritated, stressed, and annoyed were you?

In this question the participants also did not change their perception. Both medians are close are just 1 point apart. The first condition scored a median of 2 and the second condition scored a median of 3. The low median could indicate that the second condition, although being more complex, did not cause any stress or irritation.

As with the SUS the overall results of the two conditions regarding the NASA TLX scale are very similar. Which leads to the assumption that maybe an even more complex system in the future could be developed, without causing too much irritation, stress or hindering the overall performance of the user. To solve the occlusion problem, the usability of the solution plays a significant role. By solving the occlusion problem, the usability should not be hindered, which would make the solution not ideal.

8.3 Preference Questions

In the last section of the questionnaire the participants were asked to answer questions by stating their favorite condition.

QP1: Which condition did you prefer the most?

For the first question, 4 participants preferred the second condition with the smartphone, as 3 preferred the first condition and 3 the third condition. Although not by a big difference, the second condition with the smartphone is preferred by most of the participants. Keeping the research question in mind, although by a small margin, the solution to mirror the visualization seems to be preferred alternative.

QP2: Which condition was most enjoyable?

In contrast to 8.3, the participants found that the most enjoyable condition was the first one. While the first conditions got 6 votes, the other conditions could only gather 2 votes each. It can be assumed that the added complexity and effort of the conditions 2 and 3 are not justifying the information gained. A big factor for the lower score of the third condition, could also be the lack of comfort of the Google Cardboard. In the additional section for the comments in 7 this was one of the criticism of the condition.

QP3: Which condition was most helpful?

In the third question it is interesting to see that the third condition with the AR-visualization is the best among the three with 5 votes. Followed by the second condition with 4 votes. The first condition gathered only 1 vote. This is a big contrast to the results in 8.3. The big difference in the votes of the first condition indicates, that even though the usage of only the smartwatch was easier and thus more enjoyable, the occlusion problem could have been a factor in this decision. Thus making the first solution much less helpful than the other solutions. In regard to the research question of this thesis, this could be an indicator that the added information gain, with the developed extensions, is indeed helpful for overcoming the occlusion problem and thus achieving a better rehabilitation process.

QP4: Which condition was most practical?

To support the assumption in 8.3, the next question shows that when it comes to the practicality of the presented solutions, the first condition gathered, with 4 votes, the most out of the three conditions. The conditions two and three achieved 3 votes each. Looking at the research question this is not optimal. The occlusion problem is not practical. A solution to solve this issue should not introduce new impracticalities.

QP5: Which condition do you think is more suited for the use at home?

The last question gives an interesting insight into the preferences of the participants. When it comes to the usability at home, the first condition scored the lowest out of all, with just 2 votes. The mirroring and AR condition got 4 votes each. For the use at home the participants did weigh the gained information more than the practicality.

Overall the participants seems to prefer the functionality and helpfulness of a solution rather than the ease of use. This indicates that the added information is indeed needed and plays a role in the decision on which solution would be preferred to use at home.

QR: How can we overcome the occlusion problem of guidance visualization on a smartwatch during wrist rehabilitation using a smartphone

Looking at the SUS and NASA TLX questionnaires, the results show only slight deviations between the answers of the participants. This indicates that the added complexity of the Mirroring application on the smartphone, does not affect the overall usability compared to the initial solution, only using the smartwatch. While the questionnaires for the first two conditions don't seem to differ significantly, the open questions for the AR-solution, paint a different picture. Some participants were thrown off by the AR-visualization. According to their answers, the used hardware played a significant role in this perception. Vertigo effects and dizziness during the tasks were disturbing the experience. To this must be added, that the used materials for the study, let some participants feel uncomfortable. The Google Cardboard was perceived by many participants, as uncomfortable and distracting.

Using the preference rating in 7.5, a better understanding on how the usability and the helpfulness of the systems, perceived by the participants, are related to each other. While the smartwatch alone excels in the enjoyability, it is losing its advantage when it comes to the preferred use at home and the helpfulness. Compared to both the second condition with the smartphone mirroring the visualization and the AR-approach, the first condition gathered significantly lower votes than the other two conditions. It can be assumed that the gained information during the rotation tasks, by the added components in form of the smartphone and the AR-solution, were indeed something that the participants noticed and valued. This assumption is also reflected in the answers of open-end questions by the participants.

Regarding the research question of this thesis, both of the presented solutions seem to contribute to overcoming the occlusion problem and thus delivering a more detailed and helpful rehabilitation at home.

8.4 Threats to Validity

A number of threats to validity are present in the conducted study. In this section those threats will be presented and described. Parts of the implementation and study will be addressed and their potential threats to validity will be pointed out.

As the visualizations on both conditions, the mirroring on the smartphone and the AR-implementation, rely on the data transmission from the smartwatch, a point of failure during a rotation task, is the quality of the data from the sensors of the smartwatch. Further, to be able to track the wrist rotation, the smartwatch was worn on the hand of the participants and not on the wrist. It can be assumed that the rotational data is not always accurate depending on the movement of the watch during the task. This could lead to an inaccurate tracking, which can cause the actual rotation of the wrist to be bigger or smaller than the displayed maximum angle.

An additional threat is the variety of the participants. That in the study, no affected patients were involved. This could limit the informative value of the answers that the participants gave. In addition to that this study did not include professionals with a medical background with experience in wrist rehabilitation treatment. An evaluation from an expert could give a significant amount of

insight into the implemented methods and perception of the helpfulness of the developed tools. Furthermore, only one participant did not have a technical background. The familiarity with the technology is crucial, as it affects the judgement of usability.

Looking at the study design, it is important to point out that the sequence of the tasks was the same for each participant, leaving no room for variation of the given tasks. This makes it hard to isolate specific conditions and analyze them independently of each other. It can be assumed that the judgement of the conditions could suffer because of lack of isolation for specific conditions. Leaning into that, the small sample size of the participants can influence the research findings of this work, making it vulnerable to outliers in the gathered data from the participants. The lack of the validation of an expert for the given rotation tasks can also threaten the validity of this study.

9 Conclusion and Outlook

In this thesis, two extensions of a smartwatch application were implemented and presented. The existing application measured the wrist rotation of a patient and gave feedback on the achieved results. The extensions aimed to solve the occlusion problem of the smartwatch during the rotations. For the first solution of this problem, a smartphone application was developed, which mirrors the visualization in real time. The implementation included an efficient way for the data transmission and the synchronized visualization on the smartphone. For the second solution a separate smartphone application was developed. An AR-animation was created with Blender and Unity. With the help of a Google-Cardboard the animation could be displayed in real time, showing the current status of the rotation. To verify those solutions, a study was conducted. A total of 10 participants were involved in the study. The study concluded three conditions. The first condition used just the smartwatch without any of the developed extension. For the second condition the smartphone with the mirroring application was added. In the last condition the AR-application replaced the mirroring application and the smartphone was used in combination with the Google Cardboard. The participants filled out a questionnaire regarding those three conditions. The questionnaire consisted of the SUS and the NASA TLX for the first two conditions. As the displayed information on the AR condition differed from the first two conditions, therefore the questions for it, had to be adjusted. In particular the questionnaire part for the third condition, consisted of open question regarding the AR-visualization. Concluding the study five preference questions were asked, in which the preferred condition had to be selected.

To analyze the given answers, for each of the first two conditions, every question was evaluated based on the calculated median. Additionally the preference rating votes for each condition were compared and analyzed.

It is evident that the graphs for the SUS and NASA TLX show similarities, with only small differences indicating an increasing complexity for the execution of the given tasks, thus decreasing the perceived usability of the participants. Analyzing the preference rating shows that, in perspective of the performance of the first condition, there are mainly two significant differences. One being in respect to the usability, where the first condition excels. Whereas in other categories regarding the helpfulness and functionality, the other two conditions lead, which might be caused by the constant availability of the rotational information.

Future work could focus on the further development of the smartwatch application as well as the smartphone application. Especially the usability of the smartphone application could be improved, in particular, various participants complained about the necessity to interact with the smartwatch to start a new rotation. The possibility to start a rotation from the smartphone could be implemented in the future. To gain more expressive insights, the number of participants could be increased in further work, granting the diversity of the participant group, with focus on age, medical condition

and technical expertise. In addition, better hardware especially for the AR-section of the study would improve the quality of the participants answers. As uncomfortable hardware would not influence their experience.

Bibliography

- [AA14] Y. Aung, A. Al-Jumaily. “Augmented reality-based RehaBio system for shoulder rehabilitation”. In: *Int. J. of Mechatronics and Automation* 4 (Jan. 2014), pp. 52–62. doi: [10.1504/IJMA.2014.059774](https://doi.org/10.1504/IJMA.2014.059774) (cit. on p. 20).
- [ale10] alembic.io. <https://www.alembic.io/>. Last visited: December, 2023. 2010 (cit. on p. 17).
- [ASC+13] S. Ananthanarayan, M. Sheh, A. Chien, H. Profita, K. Siek. “Pt Viz: Towards a Wearable Device for Visualizing Knee Rehabilitation Exercises”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’13. Paris, France: Association for Computing Machinery, 2013, pp. 1247–1250. ISBN: 9781450318990. doi: [10.1145/2470654.2466161](https://doi.org/10.1145/2470654.2466161). URL: <https://doi.org/10.1145/2470654.2466161> (cit. on p. 19).
- [Bro95] J. Brooke. “SUS: A quick and dirty usability scale”. In: *Usability Eval. Ind.* 189 (Nov. 1995) (cit. on p. 37).
- [DCF+16] A. E. F. Da Gama, T. M. Chaves, L. S. Figueiredo, A. Baltar, M. Meng, N. Navab, V. Teichrieb, P. Fallavollita. “MirrARbilitation: A clinically-related gesture recognition interactive tool for an AR rehabilitation system”. In: *Computer Methods and Programs in Biomedicine* 135 (2016), pp. 105–114. ISSN: 0169-2607. doi: <https://doi.org/10.1016/j.cmpb.2016.07.014>. URL: <https://www.sciencedirect.com/science/article/pii/S0169260716300542> (cit. on p. 20).
- [DMP09] D. Deponti, D. Maggiorini, C. E. Palazzi. “DroidGlove: An android-based application for wrist rehabilitation”. In: *2009 International Conference on Ultra Modern Telecommunications and Workshops*. 2009, pp. 1–7. doi: [10.1109/ICUMT.2009.5345442](https://doi.org/10.1109/ICUMT.2009.5345442) (cit. on p. 20).
- [FLV09] G. D. Ferrier, H. Leleu, V. G. Valdmanis. “Hospital capacity in large urban areas: is there enough in times of need?” In: *Journal of Productivity Analysis* 32.2 (Oct. 2009), pp. 103–117. ISSN: 1573-0441. doi: [10.1007/s11123-008-0099-1](https://doi.org/10.1007/s11123-008-0099-1). URL: <https://doi.org/10.1007/s11123-008-0099-1> (cit. on p. 15).
- [Gax20] Gaxil. *Big Idea: Baking*. <https://github.com/Gaxil/Unity-AlembicToVAT>. Last visited: December, 2023. 2020 (cit. on p. 26, 32).
- [HS88] S. G. Hart, L. E. Staveland. “Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research”. In: *Human Mental Workload*. Ed. by P. A. Hancock, N. Meshkati. Vol. 52. Advances in Psychology. North-Holland, 1988, pp. 139–183. doi: [https://doi.org/10.1016/S0166-4115\(08\)62386-9](https://doi.org/10.1016/S0166-4115(08)62386-9). URL: <https://www.sciencedirect.com/science/article/pii/S0166411508623869> (cit. on p. 37).

- [KAGE11] A. Karime, H. Al-Osman, W. Gueaieb, A. El Saddik. “E-Glove: An electronic glove with vibro-tactile feedback for wrist rehabilitation of post-stroke patients”. In: *2011 IEEE International Conference on Multimedia and Expo*. 2011, pp. 1–6. doi: 10.1109/ICME.2011.6012252 (cit. on p. 19).
- [Ken23] Kent Trammell. *Big Idea: Baking*. <https://cgcookie.com/posts/big-idea-baking>. Last visited: December, 2023. 2023 (cit. on p. 16).
- [KPL+14] T. S. Kim, D. D. H. Park, Y. B. Lee, D. G. Han, J. s. Shim, Y. J. Lee, P. C. W. Kim. “A Study on the Measurement of Wrist Motion Range Using the iPhone 4 Gyroscope Application”. In: *Annals of Plastic Surgery* 73.2 (2014). issn: 0148-7043. url: https://journals.lww.com/annalsplasticsurgery/fulltext/2014/08000/a_study_on_the_measurement_of_wrist_motion_range.21.aspx (cit. on p. 15).
- [NRM+18] K. Nizamis, N. H. M. Rijken, A. Mendes, M. M. H. P. Janssen, A. Bergsma, B. F. J. M. Koopman. “A Novel Setup and Protocol to Measure the Range of Motion of the Wrist and the Hand”. In: *Sensors* 18.10 (2018). issn: 1424-8220. doi: 10.3390/s18103230. url: <https://www.mdpi.com/1424-8220/18/10/3230> (cit. on pp. 13, 15).
- [Pos80] J. Postel. “User Datagram Protocol”. In: *RFC 768* (1980), pp. 1–3. url: <https://api.semanticscholar.org/CorpusID:32952415> (cit. on p. 17).
- [RØK+16] R. S. Rasmussen, A. Østergaard, P. Kjær, A. Skerris, C. Skou, J. Christoffersen, L. S. Seest, M. B. Poulsen, F. Rønholt, K. Overgaard. “Stroke rehabilitation at home before and after discharge reduced disability and improved quality of life: a randomised controlled trial”. In: *Clinical Rehabilitation* 30.3 (2016). PMID: 25758941, pp. 225–236. doi: 10.1177/0269215515575165. eprint: <https://doi.org/10.1177/0269215515575165>. url: <https://doi.org/10.1177/0269215515575165> (cit. on p. 13).
- [TSM12] T. Than, A. San, T. Myint. “Biokinetic Study of the Wrist joint”. In: *International Journal of Collaborative Research on Internal Medicine and Public Health* 4 (May 2012) (cit. on p. 15).
- [VS08] N. G. Vinson, J. Singer. “A Practical Guide to Ethical Research Involving Humans”. In: *Guide to Advanced Empirical Software Engineering*. Ed. by F. Shull, J. Singer, D. I. K. Sjøberg. London: Springer London, 2008, pp. 229–256. isbn: 978-1-84800-044-5. doi: 10.1007/978-1-84800-044-5_9. url: https://doi.org/10.1007/978-1-84800-044-5_9 (cit. on p. 37).
- [WSO+17] D. Wenig, J. Schöning, A. Olwal, M. Oben, R. Malaka. “WatchThru: Expanding Smartwatch Displays with Mid-Air Visuals and Wrist-Worn Augmented Reality”. In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI ’17. Denver, Colorado, USA: Association for Computing Machinery, 2017, pp. 716–721. isbn: 9781450346559. doi: 10.1145/3025453.3025852. url: <https://doi.org/10.1145/3025453.3025852> (cit. on p. 19).

All links were last followed on December 12, 2023.

Declaration

I hereby declare that the work presented in this thesis is entirely my own and that I did not use any other sources and references than the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted copies.

place, date, signature